



- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*
- For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.*

IMAGE-BASED OBJECT IDENTIFICATION

I. DESCRIPTION

I.A. Related Applications

- [1] The application claims priority from a co-pending U.S. Provisional Patent Application Serial No. 60/299,734 filed June 22, 2001, the contents of which are incorporated herein by reference.

I.B. Field

- [2] A visual system and method for object identification that enables users utilizing an imaging device to obtain information about, select, purchase, or perform other operation on objects. The use may be for commerce, but is not limited to the context of commerce. The object will be identified by the image or image sequence captured with the imaging device.

I.C. Background

1. Introduction

- [3] Technology has brought with it numerous methods for performing payments and commerce-related operations in the off-line world, and for gaining information about items in a commercial or non-commercial context. In the prior art, there is a separation between methods for selecting objects

about wish to gain information, methods of making purchases, and methods of payment. The present invention can replace the existing methods gaining information, for purchasing, and/or for making payments, or may combine two or all of these methods, all based on the visual identification of objects and algorithms for processing information related to such visual identification.

2. Examples of Prior Art Product Selection Methods

- [4] The following examples are prior art methods that may be enhanced and improved by application of the invention.

a) Hand held barcode scanners:

- [5] In this case, barcode scanners are attached to a computer, PDA, cellular phone, or some such similar user device. The user scans the desired product with a barcode scanner, then the product code is extracted and used to identify the product for performing a commerce related operation such as buying the product.

b) Selection from a catalog:

- [6] A buyer marks on a catalog, or a special form, which items and what quantities he/she wants to order, and sends a fax of the document as an order form. The fax may be embedded as an attached file in an email to the seller of the item.

3. Examples of Prior Art Payment Methods

- [7] The following examples of prior art methods may be enhanced and improved by application of the invention.

a) Cellular phones and wireless PDA's:

- [8] Cellular phones and wireless PDA's can be used for performing payments with the proper e-wallet software for transferring the credit card number, authenticating the user and verifying his password and/or performing biometric tests.

b) Credit cards and smart cards

- [9] Credit cards and/or smart cards are used as proof of identity for performing payments.

4. What's the problem

- [10] The main problems with conventional methods of obtaining information about objects, selecting products, and making payment are:
- [11] 1. Identification of objects is not automated in many cases, and is therefore limited to situations in which the seller or other information provider has provided a specific object reference to off-line information or other means of obtaining information about the object.
- [12] 2. Performing the payment operation requires a cumbersome procedure, such as the use of a credit card, which involves manual swiping, waiting for

the transaction to complete, and then signing or entering a PIN code on a terminal.

- [13] 3. Payments using a cellular phone or PDA, can be performed by having the unit communicate wirelessly using IR, Bluetooth, acoustic signals or cellular network wireless protocols such as GSM, CDMA, etc. The checkout unit must then include a communication device that increases costs for the retailer and requires installation.
- [14] 4. Alternatively, payments or product selection using a cellular phone or PDA can be performed by having the user enter into the device a phone number, a web address or some other access code which is marked on the checkout unit (and/or the product identification code) and then establish a wireless connection to a remote payment management unit. The disadvantage of this method is that the user has to enter a relatively long number/code by typing, speaking or writing (e.g., with a stylus), a process that is cumbersome and error prone.

II. SUMMARY

- [15] To realize the advantages discussed above, the disclosed teachings provide:

- [16] 1. An imaging device, capable of capturing one-dimensional or two-dimensional images of objects.

- [17] 2. A device capable of sending the coded image through a wired/wireless channel to remote facilities.
- [18] 3. Algorithms and software for processing and analyzing the images and for extracting from them symbolic information such as digits, letters, text, logos, symbols or icons
- [19] 4. Algorithms and software facilitating the identification of the imaged objects based on the information gathered from the image and the information available in databases.
- [20] 5. Algorithms and software for offering various information or services to the user of the imaging device based on the information gathered from the image and the information available in databases.
- [21] The present invention relates generally to image based object identification, and more specifically to a visual method for object identification. The invention enables users that utilize an imaging device to obtain information about, select, purchase, or perform other operation on objects. Each object will be identified by the image or image sequence captured with the imaging device.
- [22] Typical activities enabled by the invention are:

- [23] 1. Selecting an object for inquiring more information about it, e.g., requesting an independent review of a book one encounters in a book-store, medical information about a drug, etc.
- [24] 2. Selecting an object to add it to a virtual "shopping cart", so that the user may go through a store, quickly adding products to such a cart and then ordering them on the spot or at a later time, when reviewing the total order made.
- [25] 3. Selecting an object for performing comparison-shopping, finding the various prices for the same article of commerce by various on-line and off-line stores.
- [26] 4. Performing the payment operation by pointing at a properly marked cash register/check out counter or label. Information read by user device (which may be a mobile or fixed handheld unit, a personal computer or some such other device) will then be used to determine the type of service or product to be purchased, and the payment required.
- [27] 5. Selecting an object such as a printed discount coupon or an article of commerce, in order to receive a discount on purchase of the item.
- [28] 6. Selecting a reference for a physical object (e.g., a picture of the object, its name, its product code, or an advertisement of the product) and then performing any or all of the operations detailed in 1-5. For example, the user

may see an item of food, then price, order, and pay for the food, all based on the visual identification of the object and the algorithms associated with the invention.

[29] 7. Selecting an object to add it to a memory. For example, the details of a business card are added to a personal phone book. As another example, the user may add the picture of a home to memory, and later use that for comparison shopping with pictures of other homes.

[30] 8. Selecting an object for registration, e.g., selection of a service that requires the transmission of the user's details like joining a member's only club or participating in a lottery.

[31] 9. Selecting an object for initiation of a phone call, connecting to a web-site, etc.

[32] The present invention, described here, solves all of these problems by enabling users utilizing an imaging device to obtain information about, select, purchase, or perform other operations on objects/products in the context of commerce or in non-commercial contexts. The object will be identified by the image or image sequence captured with the imaging device. For the act of purchasing, a credit card number transmitting can be based on the identification of the wireless device owner. Thus, the need to enter long and confusing codes or access numbers is eliminated, and at the same time robust

positive identification (including image storage as proof of purchase in cases of dispute) is provided.

- [33] In these respects, the visual method for object identification in commerce according to the present invention substantially departs from the conventional concepts and designs of the prior art, and in so doing provides an apparatus primarily developed to enable users utilizing an imaging device to obtain information about, select, purchase, or perform other operation on objects in the context of commerce or in non-commercial contexts. The object will be identified by the image or image sequence captured with the imaging device.

III. BRIEF DESCRIPTION OF THE DRAWINGS

- [34] The above objectives and advantages of the disclosed teachings will become more apparent by describing in detail preferred embodiments thereof with reference to the attached drawings in which:
- [35] Figure 1 illustrates an aspect of the invention. The selection of an object for the purpose of inquiring more information about it.
- [36] 1.1 A person is in a specific store (which may be physical or virtual).
- [37] 1.2 The person takes an image of a specific book with a mobile phone (or a PDA etc.).
- [38] 1.3 The image (or parts of it) is sent to a remote server. The image can include the title, the author's name or the barcode.

- [39] 1.4 The person receives, to his mobile phone or any other personal device capable of receiving information, a review of the book from a web site or other relevant information.
- [40] Figure 2 illustrates another aspect of the invention. The selection of an object for the purpose of adding it to a virtual "shopping cart" – thus allowing the user to go through a store, quickly adding products to such a cart and then ordering them on the spot or later when reviewing the total order made.
- [41] 2.1 A person is in a store, for example, a grocery store.
- [42] 2.2 The person sees a product of interest
- [43] 2.3 The person takes an image of the product with a PDA or a cellular device. The product is added to the person's shopping list
- [44] 2.4 The person repeats actions 2.2 & 2.3 and enlarges the shopping list.
- [45] 2.5 The order is made based on the shopping list.
- [46] 2.6 The payment is made based on the shopping list.
- [47] Figure 3 illustrates another aspect of the invention. The selection of an object from a catalog, for the purpose of making a purchase order.
- [48] 3.1 A person looks at a fashion catalog
- [49] 3.2 A product is selected and photographed directly from the catalog using a cellular telephone, PDA or any other device capable of taking a picture.

- [50] 3.3 The product is added to the shopping list
- [51] 3.4 The order and payment is made based on the shopping list.
- [52] Figure 4 illustrates another aspect of the invention. Performing a payment operation by pointing at a properly marked cash register/check out counter/label.
- [53] 4.1 A person is in a parking lot.
- [54] 4.2 The person, in this case perhaps a parking lot attendant, points at a plate containing information about the payment action required and takes a picture of the plate.
- [55] 4.3 The payment is performed and the parking lot's gate is open (It will be appreciated, however, that this method can be totally automated, without any human involvement.)
- [56] Figure 5 illustrates another aspect of the invention. Uploading a coupon or other evidence of a discount, to be used later when purchasing a specific item.
- [57] 5.1 A person reads a newspaper containing a coupon
- [58] 5.2 The coupon is photographed using a cellular phone, PDA or any other device capable of taking a picture.
- [59] 5.3 The coupon is stored.
- [60] 5.4 When a product is bought the person gets a discount using the coupon directly from the cellular phone, PDA or other imaging device.

- [61] Figure 6 illustrates the flow of events for one of the above applications (purchasing through a catalog), and outlines the technical issues involved in its implementation.
- [62] 6.1 A person presses the UCnGO button on a wireless device to operate a digital camera. (It will be appreciated that the invention will operate with any device capable of taking a digital photograph.)
- [63] 6.2 The microprocessor inside the device takes a single photo using default parameters for the photographic action.
- [64] 6.3 The image is analyzed and the optimal exposure time for the next photo is determined. A second photo is taken according to the new parameters. The second photograph may be taken by the user or may be taken automatically by the imaging device.
- [65] 6.4 The part of interest (e.g. the serial number of the requested product in a catalog, the digits of a barcode, a phone number for customer services etc.) is extracted from the entire image and is compressed.
- [66] 6.5 The image is transmitted to and received by a base station.
- [67] 6.6 The image information and the identification of the user is transferred from the base station to UCnGO's servers, or other server, through an IP net, or other digital network.

- [68] 6.7 The image is processed in the server. The relevant text/digits/icons/watermarks etc. are identified using OCR (optical character recognition) or ICR (intelligent character recognition).
- [69] 6.8 The required service is performed. For example: the cell operator, another UCnGO server or a third party performs a payment operation.
- [70] Figure 7 illustrates the cross correlation map.
- [71] Figure 8 illustrates the binarized cross correlation map.
- [72] Figure 9 illustrates the center of mass
- [73] Figure 10 illustrates how the barcode candidate is rotated to become a horizontal output image.
- [74] Figure 11 illustrates results of cross correlation.
- [75] Figure 12 shows the rotated barcode.
- [76] Figure 13 shows the translated barcode.
- [77] Figure 14 shows the input image.
- [78] Figure 15 shows the result of NCC.
- [79] Figure 16 shows the output image.
- [80] Figure 17 shows the graph of the correlation of the barcode when a template is chosen.
- [81] Figure 18 illustrates the long while lines indicating where the barcode ends.
- [82] Figure 19 shows the template matching result.

- [83] Figure 20 shows the location of the bottom of the digits.
- [84] Figure 21 shows the input image.
- [85] Figure 22 shows the corrected image.
- [86] Figure 23 shows the image that is used to cut the barcode in X.
- [87] Figure 24 illustrates the finding of the barcode width used for cutting.
- [88] Figure 25 shows a raw image.
- [89] Figure 26 shows after homogenization process.
- [90] Figure 27 shows an NN initial detection.
- [91] Figure 28 shows a page scan application.
- [92] Figure 29 shows Note Messaging application.
- [93] Figure 30 shows Buying from a Catalog application.
- [94] Figure 31 shows Snap and Share application.
- [95] Figure 32 shows a paper Portal application.
- [96] Figure 33 shows an Image Processing Server.
- [97] Figure 34 shows an example System Architecture.
- [98] Figure 35 - 63 show various stages in the implementation of the barcode example discussed herein.

IV. DETAILED DESCRIPTION

- [99] The embodiment of the invention illustrated in the drawings above allows for a visual method for object identification in commerce. The system, which

comprises the following functional components, as outlined below, is described in more detail herein.

- [100] 1. An imaging device, capable of taking one-dimensional or two-dimensional images of objects.
- [101] 2. A device capable of sending the coded image through a wired/wireless channel to remote facilities.
- [102] 3. Algorithms and software for processing and analyzing the images and for extracting from them symbolic information such as digits, letters, text, logos, symbols or icons.
- [103] 4. Algorithms and software facilitating the identification of the imaged objects based on the information gathered from the image and the information available in databases.
- [104] 5. Algorithms and software for offering various information or services to the user of the imaging device based on the information gathered from the image and the information available in databases.
- [105] The imaging device captures images or video sequences, which may be processed on this device, or may be transmitted to another device for processing. The processed data is then transmitted and transferred through some kind of data network or networks to servers which process the information using the above-described algorithms, and then uses the extracted information for various applications. The servers (or other connected entities)

may then send information back through the network to the wireless device, or to other devices such as a personal computer or set-top box.

[106] The identification of the imaged object and additional information such as the user's location, preferences and/or input are used to assist in determining the operation performed (e.g., if the object is a cash register the operation is payment, if the object is a can of juice and the user is in a supermarket the operation is add to shopping cart, if the user chose the "information only" option then the operation is to send information about the imaged object back to the computer/portable device etc.) or the operation menu offered to the user. A large portion of the processing algorithms may reside on the portable device, and there may be a dynamically changing division of the algorithms running on the different parts of the system based on relative computational loads and desired user response times, changing imaging and wireless bandwidth conditions. The application software executing for a given image or image sequence may be determined based on the image content itself, rather than being fixed. The user may choose the application software based on pre-configured parameters or during the operation.

[107] The principle of operation is that using images or video sequences, a computer can decode the identity of the imaged object, for example a labeled product, a printed form, a page from a book or newspaper, a bill, a membership card, a receipt, a business card, a medical prescription etc. This

saves the user the time and effort of inputting the object identity and/or unique information pertaining to the object such as values in numerical fields, addresses in a business card, etc. This also facilitates "one click" like commerce operations on physical objects in the real world, in the sense that the user is not required to repeat the imaging capture process. The imaging device captures images or video sequences, which may be processed on this device, or processed by another device, and then transmitted and transferred through some kind of data network or networks to servers. The servers process the information using the above-described algorithms, and then use the extracted information for various applications. The servers (or other connected entities) may then send information back through the network to the wireless device, or to other devices such as a personal computer or set-top box.

IV.A. Imaging Device

[108] The imaging device is a unit capable of acquiring images, storing and/or sending them. The imaging device is a device capable of capturing single or multiple images or video streams and converting them to digital information. It is equipped with the proper optical and electro-optical imaging components and with computational and data storage. The imaging device can be a digital camera, a PDA with an internal or external camera, a cellular phone with an

internal or external camera, or a portable computational device (e.g., laptop, palmtop or web pad-like device with an internal or external camera).

IV.B. Transmitting Device

[109] The transmitting device is capable of sending images to remote facilities.

Such device may be a cellular phone, PDA, or other wireless device, but may also be a wired communication device. The transmitting device is a device capable of transferring information to remote or nearby locations. It is capable of getting the information from the imaging device for processing and transmission. It is capable of receiving information wirelessly or using a wired connection. The transmitting device can be a cellular phone, a wireless PDA, a web pad-like device communicating on a local wireless area network, a device communicating using infrared or acoustic energy, etc.

IV.C. Image Processing Algorithms / Software

[110] The image processing algorithms perform compression, artifact correction, noise reduction, color corrections, geometric corrections, imager non-uniformity correction, etc., and various image processing enhancement operations to better facilitate the operation of the next stage of image understanding algorithms. It is implemented as a plurality of software objects residing on one or more computational devices. The image processing algorithms are numerical and symbolic algorithms for the manipulation of images and video streams.

- [111] The algorithms can be implemented as software running on a general purpose processor, DSP processor, special purpose ASIC and/or FPGA's. They can be a mixture of custom developed algorithms and libraries provided by other developers or companies. They can be arranged in any logical sequence, with potential changes in the sequence of processing or parameters governing the processing determined by image type, computational requirements or outputs from other algorithms.

IV.D. Machine Vision Algorithms / Software

- [112] The machine vision algorithms perform, among other operations, digit recognition, printed and handwritten text recognition, symbol, logo and watermark recognition, and general shape recognition. The image processing algorithms are numerical and symbolic algorithms for the manipulation of images and video streams. The algorithms may reside on a different system belonging to a different entity than the image processing algorithms or the application software.

IV.E. Application Software

- [113] The system also embodies software for utilizing the information extracted in the previous computation stages for data storage, extraction and/or communication with a plurality of internal and/or external applications, such as databases, search engines, price comparison sites etc. The application software provides the overall functionality of the service, based on the

information extracted in the previous algorithmic stages. It is software for data storage, extraction and/or communication with a plurality of internal and/or external applications, such as databases, search engines, price comparison sites etc. The application software can be implemented as code running on a general purpose-processor, DSP processor, special purpose ASIC and/or FGPA's. It can be a mixture of custom developed software and libraries provided by other developers or companies. This software may reside on a different system belonging to a different entity than the rest of the system.

[114] With respect to the above description then, it is to be realized that the optimum dimensional relationships for the parts of the invention, to include variations in size, materials, shape, form, function and manner of operation, assembly and use, are deemed readily apparent and obvious to one skilled in the art, and all equivalent relationships to those illustrated in the drawings and described in the specification are intended to be encompassed by the present invention.

[115] Therefore, the foregoing is considered as illustrative only of the principles of the invention. Further, since numerous modifications and changes will readily occur to those skilled in the art, it is not desired to limit the invention to the exact construction and operation shown and described, and accordingly,

all suitable modifications and equivalents may be resorted to, falling within the scope of the invention.

IV.F. List of Algorithms Applied

- [116] The following are some of the algorithms which may be applied by the invention on the image data:
- [117] Camera Database Algorithms: 1) Optics; 2) Sensors; 3) Electronics; 4) Noise Model; 5) Compression. (All of these algorithms exist in the prior art, but their implementation as part of the present invention is new.)
- [118] OCR Algorithms: 1) Fixed Font; 2) Numerals; 3) Detect Text Regions; 4) Multifont; 5) Multilingual; 6) Dictionary Based; 7) Template Matching. (All of these algorithms exist in the prior art, but their implementation as part of the present invention is new.)
- [119] Cross Correlation Algorithms: 1) Locate Matching Points; 2) Transform to Base Template Coordinates; 3) Check Cross Correlation Peak. (All of these algorithms exist in the prior art, but their implementation as part of the present invention is new.)
- [120] Pre Processing: 1) Compression Artifact Removal; 2) Sensor Non-Uniformity Correction; 3) Geometric Corrections Pincushion; 4) Non-Linearity Correction. (All of these algorithms exist in the prior art, but their implementation as part of the present invention is new.)

- [121] Barcode ID: 1) Barcode Detection. (This algorithm is new. It is described in section 6.2.1.2 below); 2) Super-Resolved Decoding. (This algorithm exists in the prior art, but its implementation as part of the present invention is new); 3) Digit Location. (This algorithm is new. It is described in section 6.2.2.4 below.)
- [122] Color ID: 1) Adopt Illumination Model. (This algorithm, which is new, is the use of existing lighting algorithms to estimate the spectral distribution of the illumination source in the image); 2) Project Image to Base Image Color Space. (This algorithm, which is new, is the transformation of RGB coordinates of each pixel in the image to create a new image, representing the objects in the original image as they would appear under some standard reference illumination); 3) Match Color + Correlation Matching. (This algorithm is also new. Normally, cross-correlation methods work on gray level images. The novelty here is that one can perform cross-correlation where the normal scalar dot product between pixels in the template and the target image is replaced by a different mathematical operator, e.g., a scalar dot product of R, G, and B coordinates of the two pixels in the template and target image.)

IV.G. Innovative steps

- [123] 1. The identification of an object or article by a user/customer taking a picture of it, and the analysis of the image in order to extract relevant data.
- [124] 2. The processing of a digital image of an object or article in order to determine what this article is based on: text in the image, numerals in the image, barcodes in the image, logos, texture color and shape, etc.
- [125] 3. Performing the above mentioned operations where the image is taken by a digital camera.
- [126] 4. Performing the above mentioned operations where the image is taken by a digital camera that is part of a cellular phone or a PDA device or a personal computer, instant messaging device, or any other information/communication device with a digital camera that is built in or added as an attachment.
- [127] 5. Using the digital image to load a coupon that is printed otherwise stored into some form of electronic storage device.
- [128] 6. Proof of purchase – e.g., the image indicates what the person ordered.
- [129] 7. The utilization of an email message or a multimedia message in the standards defined by the MMS (multimedia messaging service) standardization group, in order to send the picture of the object to the remote server.

- [130] 8. The utilization of an image of a barcode on the object in order to identify the object by its barcode number. (See 6.2 in the Appendix for a description of the algorithms relevant to barcode decoding.)
- [131] 9. The use of text printed on the object to identify the object by using OCR, e.g., a Heinz Ketchup bottle can be identified by the words "Heinz" and "Ketchup" printed on the label.
- [132] 10. The use of location based information in conjunction with the image of the object in order to assist in the object identification and to deduce the intended operation (e.g., near the check-out counter the object may be bought, while on display the user's need may be only to obtain information about the object)

IV.H. Appendix

1. Example for cellular commerce - Connect things & TicketsAnywhere

- [133] Some examples of usage of cellular phones for paying for various services already exist. The following sections describe the concept through various examples:
- [134] Connect Things AB was launched in 1999 and later acquired and renamed "AirClic". The company develops a laser barcode reader designed to be connected to keyboards and mobile phones. Mobile phones equipped with

this device can scan barcode and supply the user with the WAP page relevant to the product being scanned.

[135] TicketsAnywhere was founded in 2000 as a spin-off from Netlight

Consulting AB. The company offers a platform for mobile ticket, vouchers, and coupons. This service allows the user to book and buy tickets via a cell phone.

[136] A partnership between Connect Things and TicketsAnywhere enables this service. Bar codes are linked to TicketsAnywhere's ticket-booking application from which ticket purchases are made. Tickets are sent via SMS to customers' cell phones, which are read, for example, via infrared light when they walk into the cinema. Using this service helps people avoid standing in long ticket lines.

[137] Wireless ticket booking is an excellent example of how easily mobile applications can be used for commerce.

2. Technical Appendix:

[138] The following sections describes the algorithms used for detecting a barcode from a live video source of relatively poor quality, similar to the type of imaging devices that would be installed in a cellular phone.

[139] The algorithm consists of 6 main steps (that will be described in details in the following paragraph):

- [140] 1. Identify the barcode in the image, by recognizing regions in the image that resemble barcodes (uniformity in one axis and change in the other, etc.) regardless of the image rotation, the tilt of the image plane to the camera and the scale (to a reasonable extent).
- [141] 2. Based on the above identification, recognize the dimensions, orientation and location of the barcode.
- [142] 3. Extract a normalized image strip of the digits accompanying the barcode and correct the image removing any geometric distortions caused by camera pose of internal camera attributes.
- [143] 4. Read the digits in the extracted strip, achieving improved quality by utilizing the barcode specific information: relative location of digits, fonts, barcode checksum.
- [144] 5. Combining the OCR results with a direct optical reading of the barcode's lines, using super-resolution, for increased accuracy of barcode detection.
- [145] 6. Invoking an application specific operation, based on the identified product ID (e.g., presenting the web page for this product)

a) Extract Barcode

- [146] This function searches to find potential barcode areas in the image. It then evaluates the angle by which the barcode is rotated and rotates the image by

that angle. This function also translates the rotated barcode image, so that it coincides with the image center. The detected barcode is then tested to determine the likelihood that positive product identification can be made from this image. The entire process is repeated in video frame rate, allowing the algorithm to choose the frame most suitable for the detection of the barcode digits.

(1) Image Enhancement Algorithms.

[147] These functions are a family of image processing functions required in order to improve contrast and resolution, for other image processing algorithms.

(2) Finding Barcode Areas in the image

[148] The image is divided into square regions, 32x32 pixels in size. On each of these regions, a template is created by taking an 8x8 pixel array from the center of the region. Using cross correlation between each region and its corresponding template, a correlation map is formed. This map is bright for those regions where the template was found. Due to the barcode property of consisting of a series of parallel lines, the correlation map assumes a very specific shape for those areas which are barcode. The correlation map looks like a very bright line over a dark background. The angle in which the line is slanted is the angle in which the barcode lines appear in the image. Regions that are part of the barcode result with long narrow white lines, while other

regions may have a small bright spot in the region center or have a large bright/dark areas that are not narrow and long. Figure 7 illustrates a correlation map of a typical barcode image.

- [149] The aspect ratio of these correlation lines is checked and regions that contain narrow and long correlations are labeled as potential barcode candidates. Each 32x32 region is given a gray level that corresponds with how good the correlation aspect ratio is for that region. This image is then binarized with an absolute threshold, leaving the potential barcode areas white, and all other areas black, as illustrated in Figure 8. There may be several disconnected white regions in this image. Using some morphological operations, internal holes are closed, and isolated areas are removed, leaving only the large connected areas that are potential barcode candidates.
- [150] These blocks are labeled, and checked further to determine the barcode location in the image. For each block the center of mass is calculated, as illustrated in Figure 9.
- [151] The width of each block is then determined by detecting the left and right edges of the barcode candidate block. This data will later be used in order to rescale the image, as is required by the digit recognition algorithms. In Figure 10, three round dots are placed over the center of mass, as well as the right and left edges of the block. (for demonstration purposes only)

(3) Rotating the barcode

[152] This function computes the barcode's rotation angle with a maximal error of 1 degree by performing a normalized cross correlation operation. The template is a 32x32 block, taken from the best barcode candidate centroid. Figure 11 shows the result of this cross correlation. The angle of this line is calculated by a least mean square approximation to a line. The best line fit is found, and its angle is computed.

(4) Find barcode height and digit size

[153] This function finds the barcode height and digit size. This data is required in order to rescale the barcode to a proper scale, as required by the OCR algorithms. The barcode digit size is found by detecting the lower edge of the barcode lines as the upper digit delimiter, and the detecting the white strip that is under the digits, as a lower digit delimiter. The barcode's lower edge is detected by finding the cross correlation image between the barcode itself, using a wide template taken from the upper barcode part, 1x64 pixels in size. Figure 14 and Figure 15 show the raw barcode image and the result of this cross correlation, respectively. This correlation is very high inside the barcode area due to the nature of the barcode being a series of parallel lines. The correlation drops abruptly when the template is moved below the bottom barcode edge, as illustrated in Figure 17. This correlation drop, seen as the

end of the white line in Figure 15, is the barcode bottom edge, and is also shown overlaid on the raw image in Figure 16.

- [154] A similar method of template matching, with a wide white strip is used to detect the digit bottom delimiter. Figure 18 is a result of increasing the contrast with image homogenization, and binarizing Figure 16. The long white lines are enhanced by a wide high pass filter. This step is meant to emphasize the white line at the bottom of the digits. The template matching is now applied, by using a 1x64 white strip as the template, to achieve a templated matching result (as in Figure 19). Because the template is white, its correlation with the digit section is very low. However, when the template is no longer over the digit area, a high correlation appears. This can be guaranteed, as the barcode standard requires a white space under the digits. Figure 20 shows the bottom of the digits marked with a circle.

(5) Correcting Geometric deformations

- [155] This function applies an affine transformation to correct the geometric deformation in the barcode, caused by a camera viewing angle that is not perpendicular to the barcode plane. This deformation correction is required for algorithms, which are applied later in the processing chain. Figure 20 illustrates the effects of placing the camera so that the optical axis is not normal to the barcode plane. Although the barcode has already been rotated, so that the barcode lines are vertical in the image, the digits are not horizontal.

This is due to the perspective effects in the image, as different barcode points have different depth values in the camera coordinate system. The white line in Figure 21, which depicts the digit bottom line, allows for the calculation of the transformation required, in order to warp the image so that the digit centers are on a horizontal line. Figure 22 shows the corrected image, after the above transformation has been applied.

b) Digit Detection

- [156] The above processing procedures, find the barcode, and measure it's location in the frame, as well as its rotation angle, and perspective distortion. This process is done for a series of frames, in order to automatically find the most suitable frame for digit detection. When such a frame is found, it is passed on to the digit detection phase as described in the following sections.

(1) Extracting the digit section from the frame

- [157] This section describes the process of extracting the region, which contains the barcode digits, from the frame, given the barcode location. Although barcode width has already been measured, barcode edges are measured again, more accurately, in order to remove image sections on the right and left of the barcode. In order to do this, rectangular areas, five pixels in height and two pixels in width, are averaged, to form Figure 23. This is done, by performing a blurring operation for a window 5x2. As a result of averaging some bottom

lines of the averaged strip become uniformly gray. These lines are later ignored.

[158] We now utilize the fact that the vertical white space between the last bar and any other object in the image must be wider than the maximal space allowed between bars inside the barcode area. We search for the horizontal edges of the barcode by fitting a polynomial to the gray level of column vectors in the image. In order to do this, we average 3 or 4 column vectors to form a column vector, 1 pixel in width. We fit a line to these values and find a mean deviation from it. This process is repeated for each column vector in the image. The result of this, is a vector, the size of the image's width, containing the values of the mean deviations calculated above.

[159] These values are high in the barcode area and fall abruptly at the edges, as illustrated in Figure 24. This method is very accurate, and allows the removal of images sections outside of the barcode.

[160] The resulting image is also cut from above and below the digits, using the data that was previously measured, of the barcode bottom edge and the digit lower delimiter. The resulting image can be seen in Figure 25.

(2) Image Enhancement

[161] Various image enhancement techniques are used to emphasize only the relevant data of the barcode, while suppressing background noise. Figure 25 is what the image looks like after having cut the sections around the digits.

[162] Now we correct the light non-uniformity in the image, by setting the max/min values of the image to be 0.2 at black side and 0.7 at white. We then perform homogenization, binarization and remove the black line at the bottom (which usually appears on images of barcodes taken from dark objects). The final output is a binary image of the strip, as seen in Figure 26.

(3) Applying the Neural Networks for digit detection

[163] After having cut the digit section from the rest of the image, we run a digit recognizing neural network on it. Cutting the image is required in order to minimize the CPU time of this section, as well as avoiding the detection of digits which are not part of the barcode. We apply the two different neural on the whole final strip and find the activations coordinates. These neural networks are different in that they were trained with different data sets, one slightly more restrictive than the other. We then combine the activations of both networks and consider only those points, which were found by both of them. This step is made to avoid false activations of one of the networks. The result is the detection map as seen in Figure 27. The detections are the blue dots roughly around the center of each digit.

(4) Modeling the digit position

[164] Given the initial activation points of the nets, we now find the formula for a straight line on which the digits should be on. This is done by finding the parameters of a linear equation that minimizes the error from the potential

detects found in the previous section. When this model is made, we apply the neural networks again, this time only in the pixels that immediately surround the suspected activation point.

- [165] Reference is made to Figs. 28-33. Fig. 28, is entitled, "Page Scan." In this application, the user captures one or more images of parts or of the whole of a printed document. The images are uploaded to the UCnGo server, where the images are first enhanced, and then stitched to form one large image. This large image can then be formatted for display and printing in various devices, such as fax machines, e-mail attachments, graphical file formats, etc.
- [166] Fig. 29 is entitled, "Note Messaging." In this application, the user captures one or more images of parts or the whole of some handwritten note or page. The images are uploaded to the UCnGo server, where the images are first enhanced, and then stitched to form one large image. This large image is then processed with special enhancements developed to make handwritten text (regardless of the color of the ink or the color of the background) more legible in the different display and print formats. This large image can then be formatted for display and printing in various devices, such as fax machines, e-mail attachments, graphical file formats, etc.
- [167] Fig. 30 is entitled, "Buy from Catalog." In this application, the user captures one or more images of parts or of the whole of some product. The images are uploaded to the UCnGo server, where the images are first

enhanced and then stitched to form one large image. This large image is processed to locate special marks or signs, and then to perform OCR on numerals or letters identified by these special marks (e.g., note the bull's-eye marks in figure). These numerals or letters are then used to search in a database and to identify uniquely a product that the user wants to add to his or her shopping list.

[168] Fig. 31, is entitled, "Snap n Share." This application is in essence similar to the application "Page Scan" described above, except that here the image to be captured is a note on a billboard, a sign, or some other written communication other than a page.

[169] Fig. 32 is entitled "Paper Portal." In this application, the user captures one or more images of parts or of the whole of some newspaper or magazine. The images are uploaded to the UCnGo server, where the images are first enhanced, and then stitched to form one large image. This large image is then processed to locate headlines, special symbols, etc., which typically appear in a magazine or a newspaper. The OCR is performed on these particular sections of the large image. The decoded text is then used to search in a database and to identify uniquely a news story, an advertisement, etc., about which the user wants to receive additional information.

[170] Fig. 33, is entitled, "UCnGo Image Processing Server," and shows a shopping application, in this particular case the purchase of a CD. In this

application, the user captures one or more images of parts or of the whole of some product. The images are uploaded to the UCnGo server, where the images are first enhanced and then stitched to form one large image. This large image is then processed to locate special marks, barcodes, text, or logos. Barcode decoding and/or OCR are then performed on numerals or letters. The numerals, codes, logos, and/or text, are then used to search in a database and to identify a product that the user wanted to add to his or her shopping list, or about which the user wanted to perform comparative shopping, or for some other reason requiring more information about the product.

[171] Fig. 34 is entitled "System Architecture." Of particular note is the box at the middle left, named "UcnGo Image Server" on the sheet. This box shows the static elements of the applicant's system, marked as "UcnGo System," including the Image Processing Server, the Application Server, the Web site intermediary, and the Billing Server. Each static element is connected to external elements by the lines indicated on the sheet. Some of the protocols by which communication is effected are also listed on the sheet.

[172] It will be appreciated that some of the key functions of the various elements portrayed on this sheet are as follows:

[173] 1) Image Processing Server: This receives digital imaging information for processing in accordance with the algorithms described herein.

- [174] 2) Application Server: This conducts load balancing and system management in accordance with a rule-based system, all as indicated on the sheet.
- [175] 3) The Web Site Intermediary: This connects to the Internet or other data network.
- [176] 4) Billing Server: This connects to billing clients, which typically will have databases with information necessary to management the billing process.
- [177] The entire system, and the methods described herein, are parts of the invention. The individual elements of the system, and communication with such individual elements, are also parts of the invention. Finally, although one particular embodiment of the connections among the static elements is portrayed on Fig.34 it will be appreciated that the elements and/or communication paths may be arranged differently without changing the basic functionality of the overall system.

IV.I. Bar code application in detail

- [178] We first locate the barcode, and get the image shown in Figure 35.

1. Adjust image

- [179] This part takes care for the light nonuniformity of the image. The input image is expected to be in uint8 format. The size of it is (240,320). Visually, the barcode in the input image is expected to be centered and touching the

upper side (approximately 40 upper pixels of the image should be the barcode area) of the image.

- [180] Maximum 30 degrees of tilt angle is permitted. That means the angle between any horizontal line on the image and the line connecting the centers of the digits should be less than 30 degrees.
- [181] We first convert the image to double format.
- [182] We now adjust the image contrast and brightness by fixing the value of the minimum and maximum gray level in a window, which we are sure, includes only a barcode area. We take an upper 40 lines of the image limited by 80 central columns. Since the size of the input image is known by (240,320), the central column is just number 160.
- [183] The described above window is shown in Figure 36.
- [184] Now we make the light correction: that means we set the max/min values of the image to be 0.2 at black side and 0.7 at white side. These numbers were chosen after the successful runs of the forward algorithm were mapped on the two-dimensional space of different black and white edges. The reason this relatively simple method works well is that a barcode has a pre-defined contrast in the specifications.
- [185] The result is shown in Figure 37.

2. Find the end of barcode(the upper side of the digits)

- [186] Our goal is to locate the upper side of the digits. We find the central column of the image and cut 60 columns around it. This area in its upper side is supposed to include the barcode as shown in Figure 38.
- [187] We transpose the result. Now the left part of the Img includes the bars of the barcode. The bars are parallel to x-axis as shown in Figure 39.
- [188] See the documentation of the corr_profile function
- [189] As a result we get the correlation profile as shown in Figure 40.
- [190] In essence corr profile we find now the mean correlation profile in the barcode area, which is in the first 10 columns.
- [191] Now we are looking at the profile vector and check when
- [192] it breaks, means where the barcode with high cross-correlation is over and the digits begin.
- [193] We define the point where the barcode ends in the Y direction as a combination of two conditions:
- [194] A. The correlation value is less than half of the mean correlation value achieved on the barcode itself (we always assume the first 10 rows contain a barcode).
- [195] B. In the 20 pixels after the assumed "break" in the barcode the correlation values should be small (minimum less than 0.15) or at least never reach high values (maximum less than 0.8). The reason for condition B is that we want to

protect ourselves from a small "break" in the actual barcode due to reflections etc.

[196] We are now looking for the angle between the bars and any vertical line on the image.

[197] See the documentation of the `bars_angle` function

[198] Then we make the fine angle correction

[199] Now we go again through the process of the end of barcode finding. After the barcode is perfectly rotated, it is good to check it again. The reason is that the method of finding the edge by cross correlation is sensitive to even small deviations from the X-axis. The treatment is identical to the first Y cut.

[200] Future improvements: one may replace all constant thresholds with relative ones dependent on the `barcode_corr` value.

[201] We cut a horizontal strip that includes digits and barcode. The strip is taken considerably wide in order not to miss the cases with large barcodes. Still, for now we have no estimation of the barcode's scale as shown in Figure 41.

[202] We want to make it possible for the demonstrator to stop the long routine in case of bad image taken, and go back to the beginning of the function `check_kb_mex_r` is checking whether Space was pressed till now. We check the output of this routine about 10 times during the program to avoid useful waiting for long functions to finish.

3. Cut the digits

- [203] In this part we use the same principle of high correlation in barcode area and very low in the rest. Input to the algorithm : horizontal strip of 60 pixels height, while it is already assumed the bars are perfectly straight(parallel) to y-axis. The strip should include the digits and the bars as well. The parameter `strip_half_width` defines number of pixels taken left and right from the center of the strip. `strip_half_width` was chosen, after the definition of the limit of the barcode width when the digits are still recognizable (100 pixels).
- [204] Now we take the first line in the above defined `middle_strip`, normalize it and calculate the correlation of it with the rest of the lines in the `middle_strip`.
- [205] Since the IPL library is sensitive to the size of the input images, we test and correct the y-size of the `middle_strip` and correct it if needs.
- [206] The result of the above operation is the `corr_matrix` as shown in Figure 42.
- [207] We now normalize the `corr_matrix` and get the result shown in Figure 43.
- [208] Since the bars are straight, we don't need to look for the line with the highest correlation, it will be just the middle column. The upper side of the digits will be at the point, when the correlation vector falls under the defined `y_up_corr_threshold`.
- [209] The correlation vector is shown in Figure 44.

- [210] Since we adjusted the originally grabbed image (stretched it's values between 0.2 and 0.7, we want now to normalize it back. This can be done with `imadjust`. The reason for the normalization is not clear either.
- [211] We take this strip (`N_middle_strip`) thinner than the middle strip. This is because sometimes the barcode is not exactly centered. We are looking now to the lower edge of the digits. Actually we are looking at the white line under the digits. The method we use is sensitive to cases when we do not include in the image additional patterns besides part of barcode and the digits, that's why we take the strip thinner . The `N_middle_strip` is shown in Figure 45.
- [212] Now we run down on the `N_middle_strip` and check every line for the deviation from the straight line we find with the `polyfit`. As it was mentioned before, we expect for the clear line beneath the digits. Color component of this line will be all very close. Thus the `polyfit` method is applicable. Obviously we start the line tests from `y_up` and lower. Sometimes the image of `N_middle_strip` can be quite smeared and thus the threshold of minimal deviation will be different. So we define 2 cases.
- [213] The deviation vector is shown in Figure 46.
- [214] Finally we find the height of the digits in the image.
- [215] This is very important since it gives us further information about the width of the barcode; We found the standard relation between these 2

magnitudes(digits height and barcode width) is quite constant and is about 19/180 accordingly;

- [216] We resize the strip to get more-or less constant and known sizes of the barcode as shown in Figure 47.

4. Homogenization:

- [217] Now again we check the y- size of the strip because of the sensitivity of the IPL library functions.
- [218] see the documentation of the IPLMEX library and l4homogenizationFP function ;
- [219] In general the following parts: homogenization as shown in Figures 48-49, binarization as shown in Figures 52-53 and no black bottom line as shown in Figures 51-52: their final output is a binary image of the strip when we also clean it from possible black line in the bottom of the strip. This line appears usually on images of barcodes taken from dark objects. Here the final results of each part will be presented. Additional image is used in this presentation to emphasize the no_black_line effect.

5. Final y-cut

- [220] Input: This section of the code takes a binarized image of the barcode area, where the barcode has already been straightened (that is the barcode Lines are parallel to the Y axis in the Matlab axis convention). Furthermore, the Y' cut

which identifies the upper edge of the digits has already been performed, and the Y cut line is roughly at the center of The binarized input image.

[221] Output: the output of the section is the binarized image of the digits Cut from below.

[222] Algorithm: we are looking for a uniform white strip that is always below the digits. We characterize this strip by being a relatively uniform (in terms of gray levels) horizontal strip. Since we have already performed binarization we do not have to worry about lighting non-uniformity: We expect this strip to be a "constant" white value, with potential small deviations due to small black "pepper" noise after the binarization. We thus recognize the strip by the small average standard deviation.

[223] Important note: we first apply (after the binarization) an averaging filter. The purpose is to reduce the effect of residual "pepper" noise while not erasing the larger effects of digits or barcode lines. We are looking ONLY on the standard deviation (and ignore the mean).

[224] Potential future improvement: The blkproc operator could be replaced with a simple averaging filter, and perhaps the averaging could be completely removed.

[225] Comments: a relatively new piece of code (January 2001) so it is not very well tested.

[226] Averaged_Image is the result of doing a blurring operation for a window as shown in Figure 54. The size of 3 pixels (height or Y) by 15 pixels (width or X).

[227] We only concentrate for our purposes on the center 50% (in the X direction) Of the image, so as to avoid problems due to small residual misalignment errors in the barcode's rotation. The deviation vector holds the values for each line cut in the X direction as shown in Figure 55.

[228] We take a horizontal cut of the image and calculate the power-8 standard deviation for this cut.

[229] We declare a detection of the white line the first time the deviation for a given X-cut is below some threshold as shown in Figure 56.

6. X-cut

[230] averaged_strip is the result of doing a blurring operation for a window of the size of 5 pixels (height or Y) by 2 pixels (width or X). As a result of averaging some bottom lines of the averaged_strip become uniformly grey. We define the number of these lines by grey_lines and ignore them later as shown in Figure 57.

7. Cut with k=3

[231] The basic assumption in the following algorithm is the following:

[232] The white vertical space between the last bar and any other pattern on the tested image is wider than such a space between bars inside the barcode area.

[233] We search for the horizontal edges of the barcode by polyfit method. That means we run from side to side of the averaged_strip, with one pixel step, take k (parameter) columns at each step, average them to get 1 column.

[234] We treat this column just as ordinary vector with numbers, while the numbers are actually grey color levels of the pixels. We fit a line to these values and find a mean deviation from it. After all we are left with a vector of length=width(strip), with high values in the barcode area and falling at the edges as shown in Figure 58.

[235] We start from the middle coordinate and move towards the edges of the vector mean_deviation, while we compare its values to a deviation_threshold_X. The first time the deviation falls below the threshold, we define as the edge of the barcode.

8. Cut with k=4

[236] Now proceed the same process, while we use the parameter k=4 this time. The reason for use of both of them, is the sensitivity of the above algorithm to the initial scale. We always get successful results with one of the values of the parameter k. Sometimes both of them work. Generally, for large barcodes k=4 statistically works better, and accordingly, for smaller barcodes k=3 performs better results. By small and large barcodes, we mean barcodes, which have been photographed from large and close distance, accordingly.

9. Choose $k=3$ or $k=4$

- [237] Now after we get two sets of coordinates of the horizontal edges of the barcode, the next task is to choose the right between them. The basic assumption is that the width of the barcode according to $k=4$ is always bigger than the one calculated with $k=3$; We apply now additional algorithm which will help us to choose the right answer;
- [238] The following algorithm uses the zero-crossing effect: If we look at the barcode area horizontal image profile, we expect for large oscillations (many mean-crossings, with small distances between them); When the barcode ends the oscillations are fading down.
- [239] We take the strip between the found coordinates and add some additional confidence step from both sides as shown in Figure 59.
- [240] We apply now the blurring operator, in the way that all the dark patterns including barcode bars become smeared. This is done by finding the minimum of overlapping blocks as shown in Figure 60.
- [241] Now we take the seventh row of the strip, where we suppose the barcodes's bars are included and subtract the average of the blurred_cut_strip to achieve the oscillations around zero as shown in Figure 61.
- [242] In the following loop we are running on every couple of close coordinates and check their product to find the zero-crossing. Whenever the zero-crossing

appears, we take its coordinate (this is parallel to X-coordinate on the blurred_cut_strip), and store it in `zc_coor` vector as shown in Figure 62.

- [243] When the following zero-crossing appears, we check whether the space between its coordinate and the coordinate of the previous zero_crossing is white. That means we check whether the profile between these points was positive. If these conditions are satisfied, we store the distance between the 2 successful zero-crossings (width of the white space) in vector `Spaces` with the positive value of the width of the appropriate space. Accordingly, if the space was black, the width of it will be stored with negative sign.
- The gentle treatment of the problem requires considering the zero-crossing occurrence also if one of the two following values of the profile were sufficient close to zero, but not really crossed it.

[244] We defined the sufficiency by `zero_threshold`:

[245] The following is the `Spaces` vector plotted against the `zc_coor` vector as shown in Figure 62.

[246] Now we apply all the above steps on the cut with `k=4` strip.

10. Analyze and choose

[247] We find the total number of the spaces. This is done by calculating the length of the vector of width of all the spaces.

[248] We are looking now for the coordinate of the middle of the barcode to proceed from there to left and right and find the edges of the barcode. Since

this coordinate is not definitely the median one in vector of zero_crossings coordinates, we make it by finding it explicitly: by finding the mean value of the `zc_coor`, and locating the closest member in `zc_coor`;

[249] We now run on vector of spaces width (with positive values for white spaces and negative values for black spaces);

[250] We expect the width of the white space indicating the end of the barcode to be larger than `space_threshold`

[251] When the above condition is satisfied, we go back to the vector of the coordinates of the zero-crossings (which are real coordinates on the `blurred_cut_strip`) and locate the appropriate coordinate of the beginning of the wide white space.

[252] Additional condition we are checking for is: a wide black space in the middle of the barcode. This is happening as a result of blurring minimizing operator we applied before. Two central bars of the barcode smear to one wide black bar. This finally occurs as large negative value in Spaces vector the middle of the barcode is taken as the middle coordinate between the left and the right coordinates found by zero-crossing method;

[253] We also check here the existence of the wide white space in left part of the `space_vector`;

[254] Finally, when all of the above are satisfied, we check whether the correction done by zero-crossing is not too large (larger than 17 pixels); If it is

larger, we forget the correction and use as final result the previously found left edge of barcode with $k=3$

[255] If none of above are not satisfied we proceed the same testings with $k=4$ parameters.

[256] After all, if none of above conditions are satisfied, we

[257] Chose as the left coordinate of the barcode, the one, previously found with $k=4$ without zero-cross correction, we believe that using $k=4$ will not cause cutting the barcode in the middle. That is why we chose this result to prevent over cutting in any case.

[258] All the above steps are hold from the right side too.

11. Final resizing

[259] Now we take our strip and cut from it the final image with the digits and minimum barcode area. We know the y-coordinate of the digits upper edge. But we should remember that this coordinates are good for the strip before the resizing we made. Taking this fact into account and adding additional 5 pixels of barcode area (in previous scaling) the y_{up} and translating the result to the recent sizes of the strip, and limiting the strip from left and right, we get the `final_strip`

[260] Now we are adjusting the sizes of the `final_strip` according to our knowledge of the barcode width, which is believed to be more exact than the digits height.

12. Apply neurals

[261] Now we apply 2 neural networks on the whole final_strip and find the activations coordinates. We then combine the activations of both networks and consider only those points, which were found by both of them. This step is made to avoid false activations of one of the networks.

[262] The result of this part is shown in Figure 63.

[263] The green points indicate the activations of the networks.

13. Format model

[264] We are fitting a model on the activation points coordinates. In the model we assume that the activations of the neurals should construct a pair of lines with 6 points in each. The distance between all the 60 points is equal, and the distance between the 2 central points is approximately double of this distance. We assume plane surface, means all the points are located on a line with the same slope.

[265] We vary the following parameters and find for such values of them, when the deviation of the real points from the predicted values is minimal.

The parameters to vary are:

[266] 1. The slope of the line connecting all the points.

[267] 2. The bias of the above line.

[268] 3. The distance between the points in each one of the set of

[269] the 6 points

[270] 4.The location of the first (left) point.

[271] 5.The distance between the central 2 points.

[272]

[273] Future improvements: The model can be adapted for curved surfaces. This is done by adding additional variable parameter of curvature of the line with the points. Also the problem of the perspective can be solved by permitting the distance between the points to be changing by a constant factor (as we move away from the center) and to vary this factor.

14. Format String

[274] We now know the final coordinates of all 12 digits in the image. The task is to recognize the digits. See the documentation of the function `better_format_string1`.

[275] Other modifications and variations to the invention will be apparent to those skilled in the art from the foregoing disclosure and teachings. Thus, while only certain embodiments of the invention have been specifically described herein, it will be apparent that numerous modifications may be made thereto without departing from the spirit and scope of the invention.

WHAT IS CLAIMED IS

1. A system for object identification that enables users utilizing an imaging device to obtain information about, select, purchase, or perform other operation on objects, comprising:

an imaging device, capable of capturing one-dimensional or two-dimensional images of objects;

a device capable of sending the coded image through a wired/wireless channel to remote facilities;

algorithms and software for processing and analyzing the images and for extracting from them symbolic information such as digits, letters, text, logos, symbols or icons;

algorithms and software facilitating the identification of the imaged objects based on the information gathered from the image and the information available in databases; and

algorithms and software for offering various information or services to the user of the imaging device based on the information gathered from the image and the information available in databases.

2. A method for obtaining information about, selecting, purchasing, or performing other operations on objects, comprising:

a) capturing an image with an image device;

- b) processing the image, which may include additional images of the object;
- c) extracting the part of the image that is of interest;
- d) compressing the data related to the part of the image that is of interest;
- e) transmitting the processed data to another location;
- f) transferring the image information and/or identification of the user to one or more servers;
- g) performing additional processing on the data at the server, including identification of that part of the image of interest;
- h) performing a required service; and
- i) communicating to a user or user device the performance of the service.

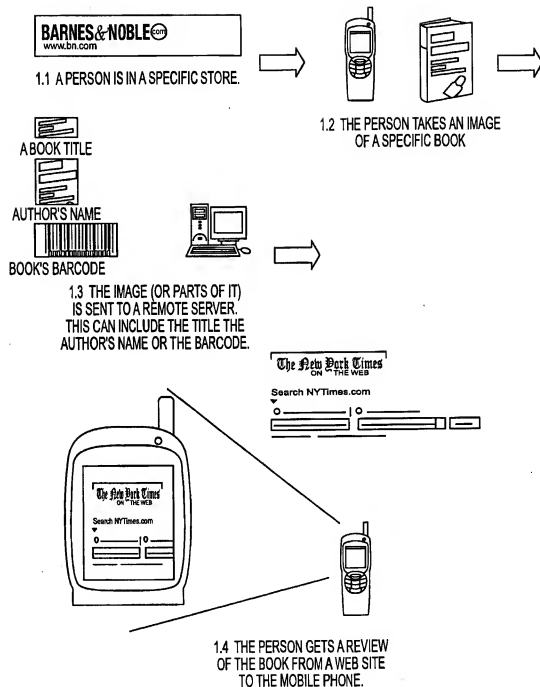
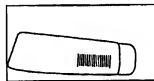


FIG. 1

 GROCERY MANUFACTURERS OF AMERICA
MAKERS OF THE WORLD'S FAVORITE BRANDS OF
FOOD, BEVERAGES, AND CONSUMER PRODUCTS



Shopping list:
empty



2.1 A PERSON IS IN A GROCERY STORE.

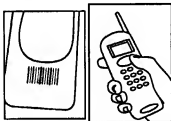
2.2 THE PERSON SEES A PRODUCT
OF INTEREST, FOR EXAMPLE, A CREAM.



Shopping list:
1. 1 Cream 5\$



2.3 THE PERSON TAKES AN IMAGE OF
THE PRODUCT WITH A PDA OR A CELLULAR DEVICE.



Shopping list:
1. 1 Cream 5\$
2. 1 Shampoo 6\$



2.4 THE PERSON REPEATS ACTIONS 2.2 & 2.3
AND ENLARGES THE SHOPPING LIST.

Order

Shopping list:
1. 1 Cream 5\$
2. 1 Shampoo 6\$
3. 4 tomatoes 2\$
4. 2 bread 1\$
5. 1 milk 1.5\$
total: 15.5\$

2.5 THE ORDER IS MADE
BASED ON THE SHOPPING LIST.



bill
total: 15.5\$

2.6 THE PAYMENT IS MADE
BASED ON THE SHOPPING LIST.

FIG. 2

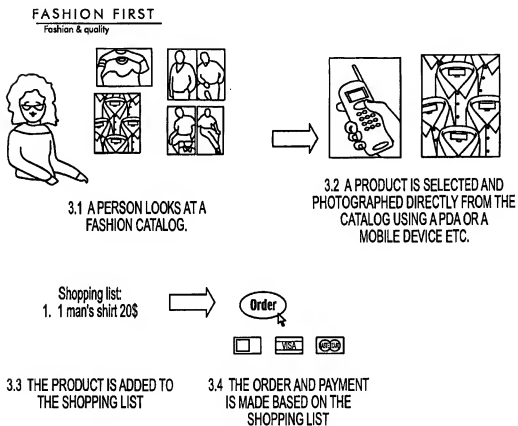


FIG. 3

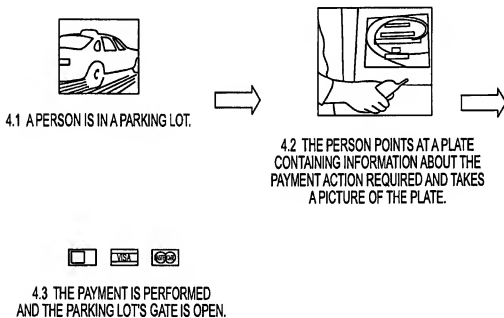
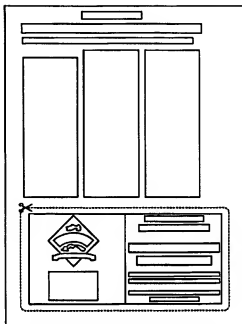
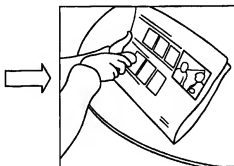


FIG. 4



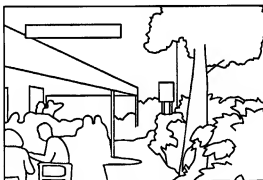
5.1 A PERSON READS A NEWSPAPER
CONTAINING A COUPON.



5.2 THE COUPON IS
PHOTOGRAPHED USING A PDA OR A
MOBILE DEVICE.



5.3 THE COUPON IS STORED.



CHEQUE

Des Champs - Restaurant
Table num. 14
Waitress: Shallot
Total amount: 37\$
Coupon discount: -5\$
For payment: 32\$



5.4 WHEN A PRODUCT IS BOUGHT THE PERSON GETS A
DISCOUNT USING THE COUPON DIRECTLY FROM THE PDA OR A
MOBILE DEVICE.

FIG. 5

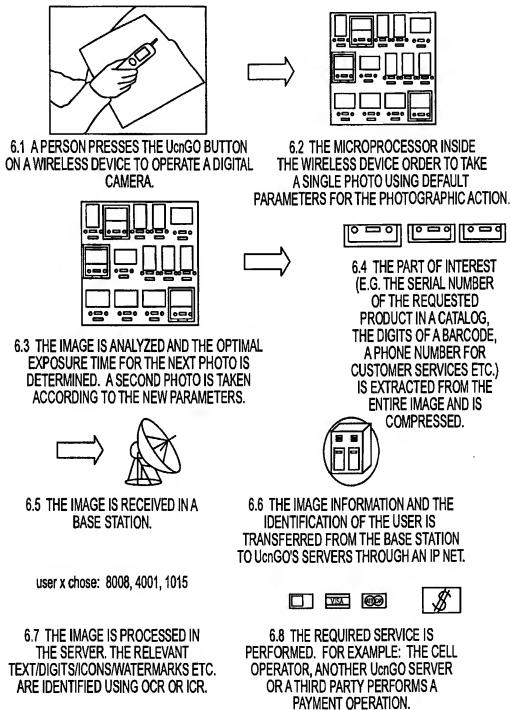


FIG. 6



FIG. 7



FIG. 8



FIG. 9



FIG. 10



FIG. 11



FIG. 12



FIG. 13



FIG. 14



FIG. 15



FIG. 16

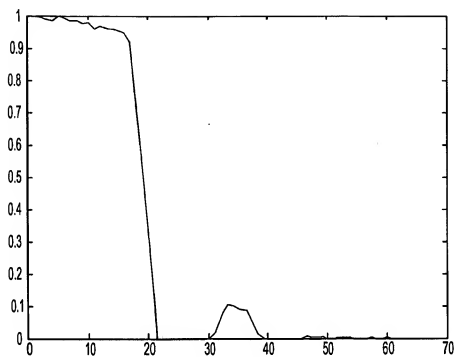


FIG. 17



FIG. 18

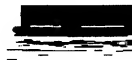


FIG. 19



FIG. 20



FIG. 21



FIG. 22



FIG. 23

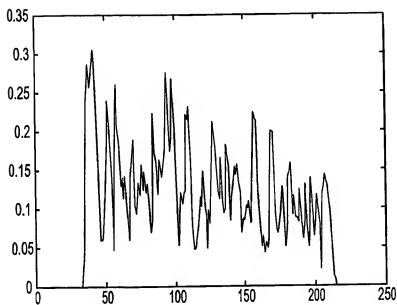


FIG. 24



FIG. 25



FIG. 26



FIG. 27

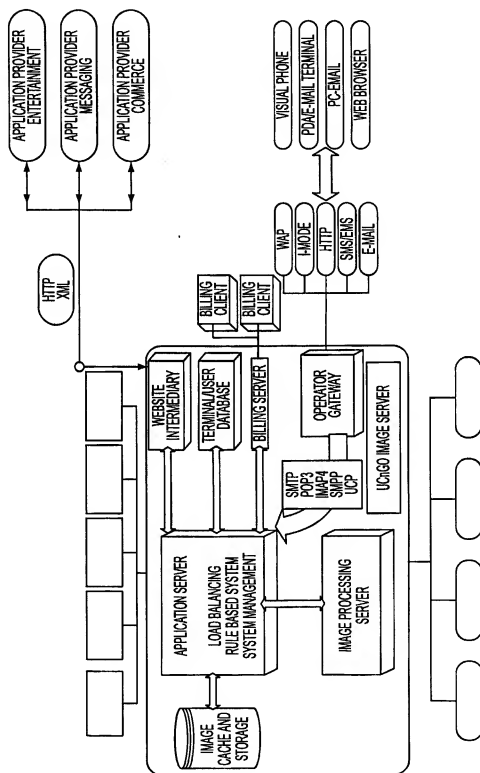


FIG. 28

BENEFITS FOR WIRELESS OPERATOR

- NEW FORM OF MOBILE MESSAGING
- "AIRTIME-HUNGRY" APPLICATIONS
- STICKY SERVICE
- M-COMMERCE ENABLING SERVICE
- ENABLES ENTERPRISE APPLICATIONS

FIG. 29

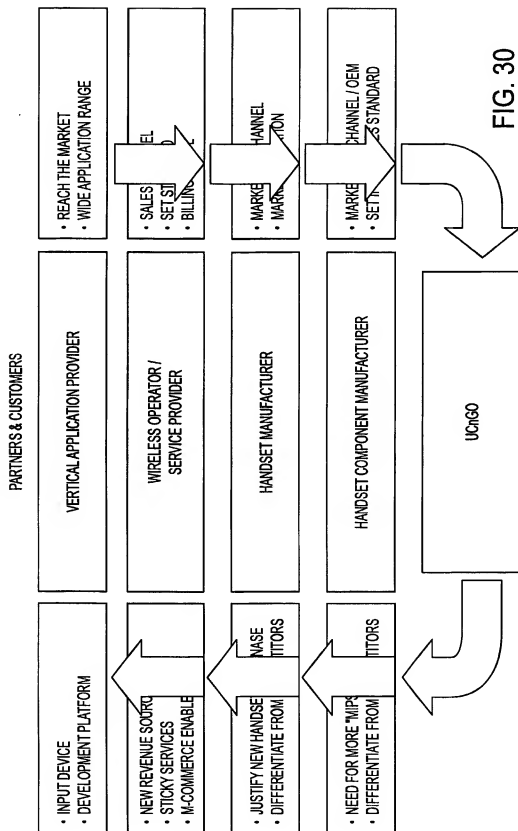


FIG. 30



COMPETITION


TYPE	PLAYER	UCrGO ADVANTAGE
WIRELESS IMAGING PLATFORMS	FLASHPOINT, LIGHTSURF	ADVANCED IMAGE RECOGNITION FEATURES, APPLICATIONS
ALTERNATIVE INPUT DEVICES	VOICE PORTALS	SERVICE QUALITY, A PICTURE IS WORTH A THOUSAND WORDS
"BARCODE" READING SOLUTIONS	AIRCLIC, CRQ, DIGIMARC	NO NEED FOR SPECIAL READER, SPECIAL CODES

FIG. 31



MARKET ENABLING TECHNOLOGIES

- CMOS IMAGERS
 - LOW COST, LOW POWER - FITS INTO PORTABLES
 - RESOLUTION - CIF, VGA
- HANDSET PROCESSING POWER
- HANDSET PROGRAMMABILITY



JAVA

 - PROGRAM IMAGER PARAMETERS
 - MMI ACROSS MULTIPLE HANDSETS
- INCREASING BANDWIDTH IN CELLULAR SYSTEMS
- PACKET DATA NETWORKS OVER CELLULAR
- COMPUTATIONAL POWER OF SERVERS

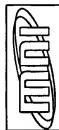
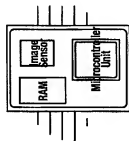


FIG. 32

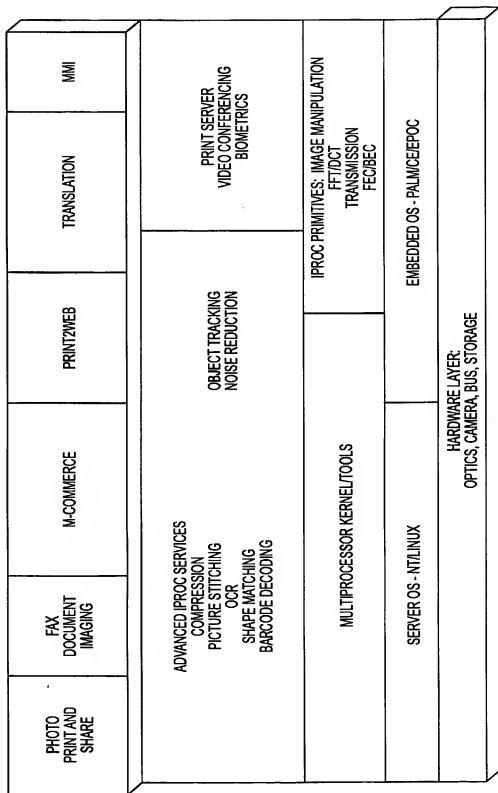


FIG. 33

R&D - KEY ACHIEVEMENTS

- ESTABLISHED FEASIBILITY FOR:
 - SENSOR
 - OPTICS
 - IMAGING ENVIRONMENT
 - COMPRESSION
- FOR IMPLEMENTATION ON CELLULAR DSP CORES AND SERVER ARCHITECTURE
- DEMONSTRATIONS FOR 3 KEY APPLICATIONS:
 - BARCODE READING
 - PRINT2WEB
 - FAX

FIG. 34

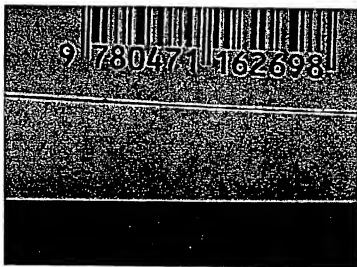


FIG. 35



FIG. 36



FIG. 37



FIG. 38



FIG. 39

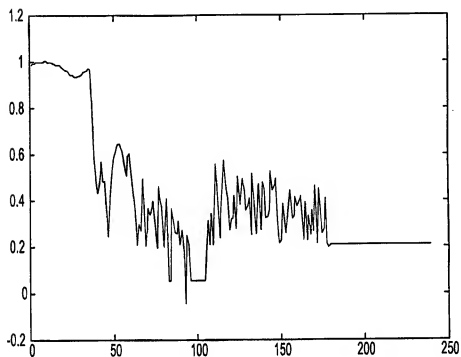


FIG. 40



FIG. 41



FIG. 42



FIG. 43

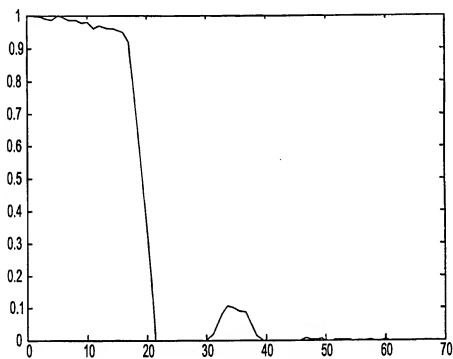


FIG. 44

||||
7.1

FIG. 45

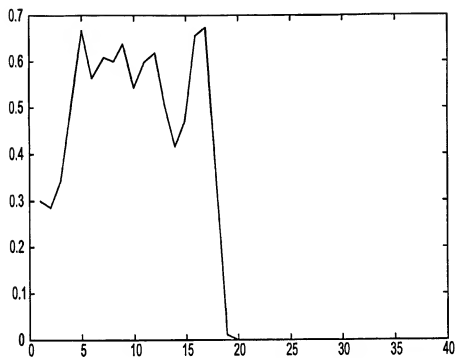


FIG. 46



FIG. 47

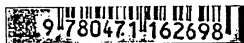


FIG. 48

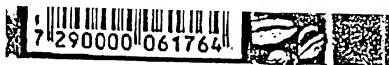


FIG. 49



FIG. 50



FIG. 51



FIG. 52



FIG. 53



FIG. 54



FIG. 55

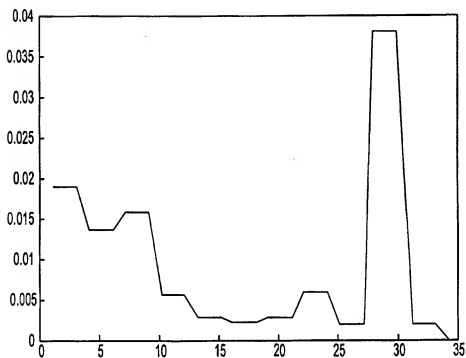


FIG. 56



FIG. 57

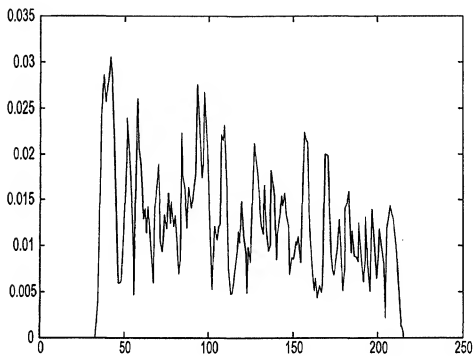


FIG. 58



FIG. 59



FIG. 60

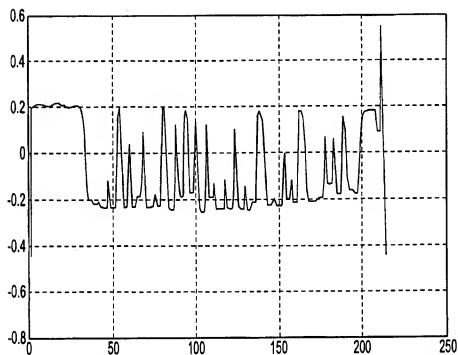


FIG. 61

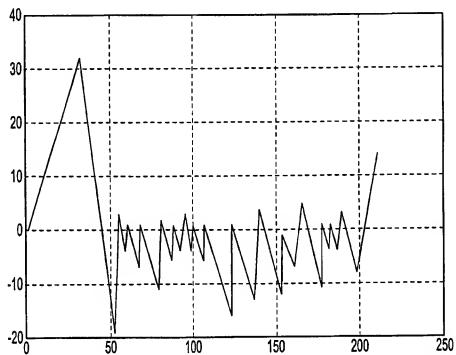


FIG. 62

9 780471 162698

FIG. 63

INTERNATIONAL SEARCH REPORT

International application No.

PCT/IB02/03352

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06K 9/00; G06F 17/60

US CL : 382/100; 348/61; 705/26

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 382/100; 348/61; 705/26

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
NONEElectronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Please See Continuation Sheet**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A,P	US 2002/0073077 A1 (LENNON) 13 June 2002 (13.06.2002), entire document.	1-2
A,E	US 2002/0082943 A1 (IBE) 27 June 2002 (27.06.2002), entire document.	1-2
A,E	US 2002/0169853 A1 (GROSSMAN) 14 November 2002 (14.11.2002) entire document.	1-2

☐ Further documents are listed in the continuation of Box C.☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

**

later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

*X

document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

*Y

document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

*Z

document member of the same patent family

Date of the actual completion of the international search

22 November 2002 (22.11.2002)

Name and mailing address of the ISA/US

Commissioner of Patents and Trademarks

Box PCT

Washington, D.C. 20231

Facsimile No. (703) 305-3230

Date of mailing of the international search report

05 DEC 2002

Authorized officer

David V Bruce

Telephone No. (703) 308-0956

Rene R. R. R.

INTERNATIONAL SEARCH REPORT

PCT/IB02/03352

Continuation of B. FIELDS SEARCHED Item 3:
USPTO EAST/BRS

search terms: image, purchase, cellular phone, database